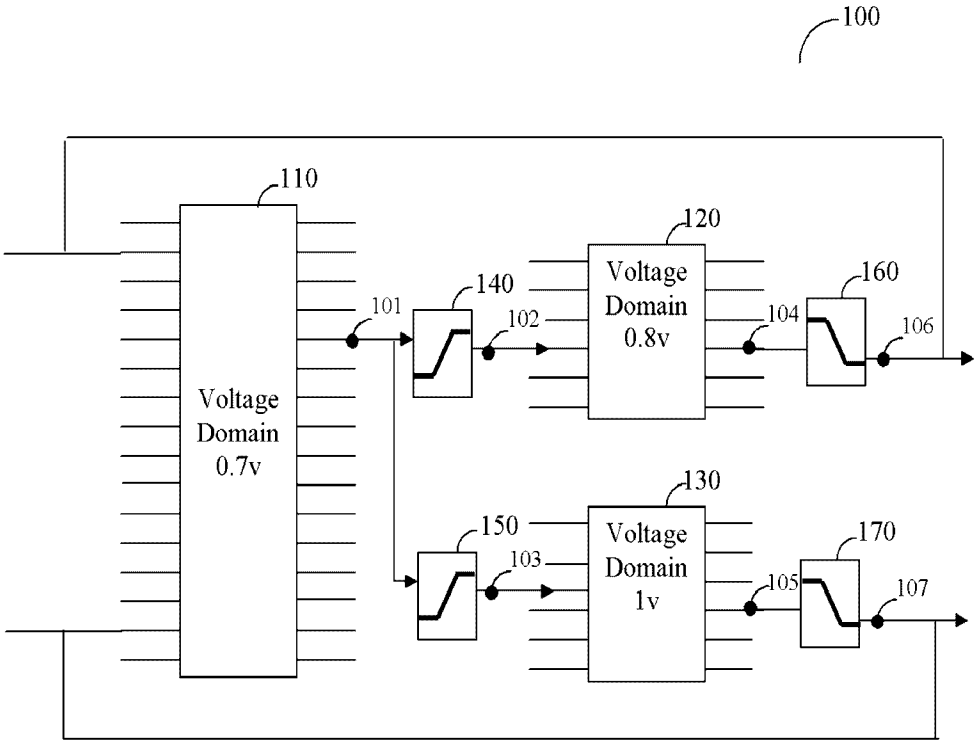

FIGURE 1 (PRIOR ART)

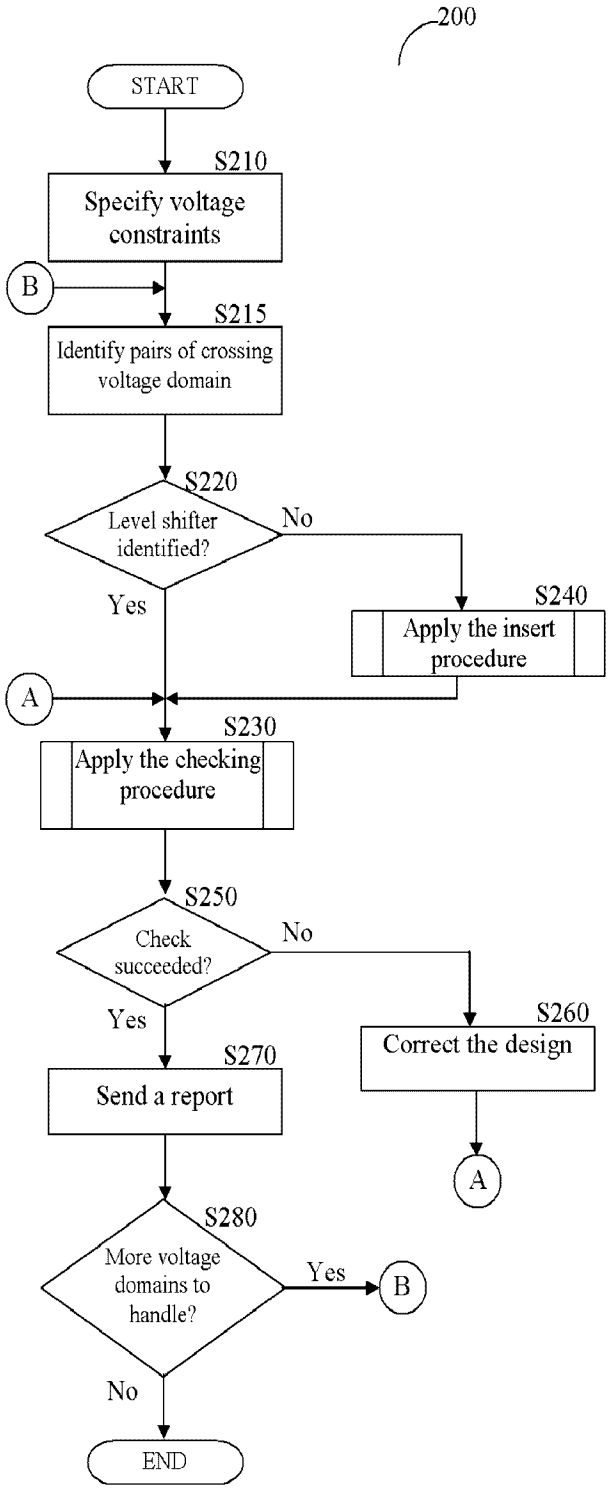
200
START
S210
Specify voltage constraints
B
S215
Identify pairs of crossing voltage domain
S220
Level shifter identified?
No
Yes
S240
Apply the insert procedure
A
S230
Apply the checking procedure
S250
Check succeeded?
No
Yes
S260
Correct the design
S270
Send a report
A
S280
More voltage domains to handle?
Yes
B
No
END

FIGURE 2

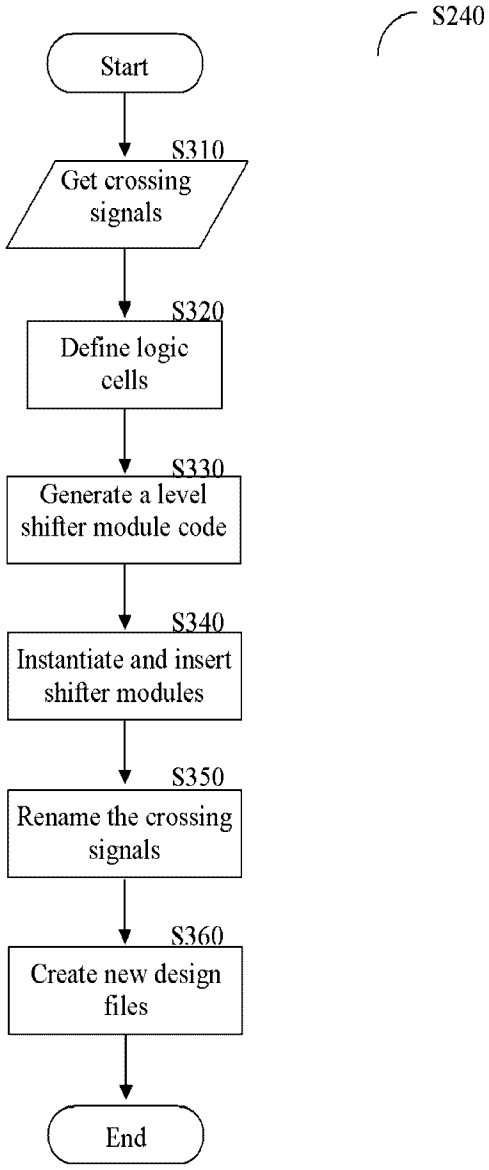
S240
Start
S310
Get crossing signals
S320
Define logic cells
S330
Generate a level shifter module code
S340
Instantiate and insert shifter modules
S350
Rename the crossing signals
S360
Create new design files
End

FIGURE 3

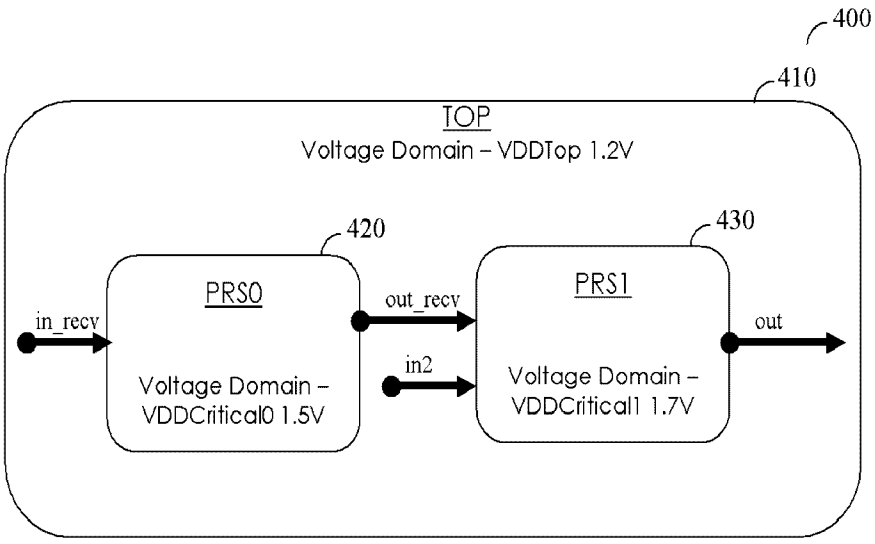
400
410
TOP
Voltage Domain – VDDTop 1.2V
420
PRS0
Voltage Domain –
VDDCritical0 1.5V
430
PRS1
Voltage Domain –
VDDCritical1 1.7V
in_recv
out_recv
in2
out

FIGURE 4A

```
module top(in1, in2, out1);

        input in1;
        output out1;
        input in2;

        wire in_recv;
        wire out_recv;
        wire enableSig;

        assign in_recv = ~in1;
        process prs0(out_recv, out_recv0);
        rec_mod  prs1(out_recv0, in2, out1);

endmodule

module rec_mod(in1, in2, out);
        input in1;
        input in2;
        output out;

        rec_process rec_prs0(in1, in2, out);

endmodule

module process(in, out);
        input in;
        output out;

        turing t0(in, out);
```

FIGURE 4B

```
4000    current_design top
4010    voltagedomain -name VDDTop -value 1.2 -modname top+
4020    Voltagedomain -name VDDCritical0 -value 1.5 -instname top.prs0+
4030    voltagedomain -name VDDCritical1 -value 1.7 -instname top.prs1+
4040    levelshifter -name LS_2X1_12TO15 -from VDDTop -to VDDCritical0 -at
VDDTop -        enableTerm EN -inTerm A -outTerm Z -enableNet enableSig
4050    levelshifter -name LS_2X1_12TO17 -from VDDTop -to VDDCritical1 -at
VDDTop -        enableTerm EN -inTerm A -outTerm Z -enableNet enableSig
4060    levelshifter -name LS_2X1_15TO12 -from VDDCritical0 -to VDDTop -at
VDDTop -        enableTerm EN -inTerm A -outTerm Z -enableNet enableSig
4070    levelshifter -name LS_2X1_17TO12 -from VDDCritical1 -to VDDTop -at
        VDDTop -enableTerm EN -inTerm A -outTerm Z -enableNet enableSig
4080    levelshifter -name LS_2X1_15TO17 -from VDDCritical0 -to VDDCritical1 -at
        VDDTop -enableTerm EN -inTerm A -outTerm Z -enableNet enableSig
```

FIGURE 4C

```
4100  module top(in1, in2, out1);

4110      input in1;
4120      output out1;
4130      input in2;

4140      wire in_recv;
4150      wire out_recv;
4160      wire enableSig;

4170      assign in_recv = ~in1;
4180      LS_2X1_12TO15 ls0(.A(in_recv), .EN(enableSig), .Z(ls0_in_recv));
4190      process prs0(ls0_in_recv, out_recv0);
4200      LS_2X1_15TO17 ls1(.A(out_recv0), .EN(enableSig), .Z(ls1_out_recv0));
4210      LS_2X1_12TO17 ls2(.A(in2), .EN(enableSig), .Z(ls2_in2));
4220      rec_mod  prs1(ls1_out_recv0, ls2_in2, ls3_out1);
4230      LS_2X1_17TO12 ls3(.A(ls3_out1), .EN(EnableSig), .Z(out1));

4240  endmodule

4250  module rec_mod(in1, in2, out);
4260      input in1;
4270      input in2;
4280      output out;

4290  rec_process rec_prs0(in1, in2, out);

4300  endmodule

4310  module process(in, out);
4320      input in;
4330      output out;

4340      turing t0(in, out);

4350  endmodule
```

FIGURE 4D

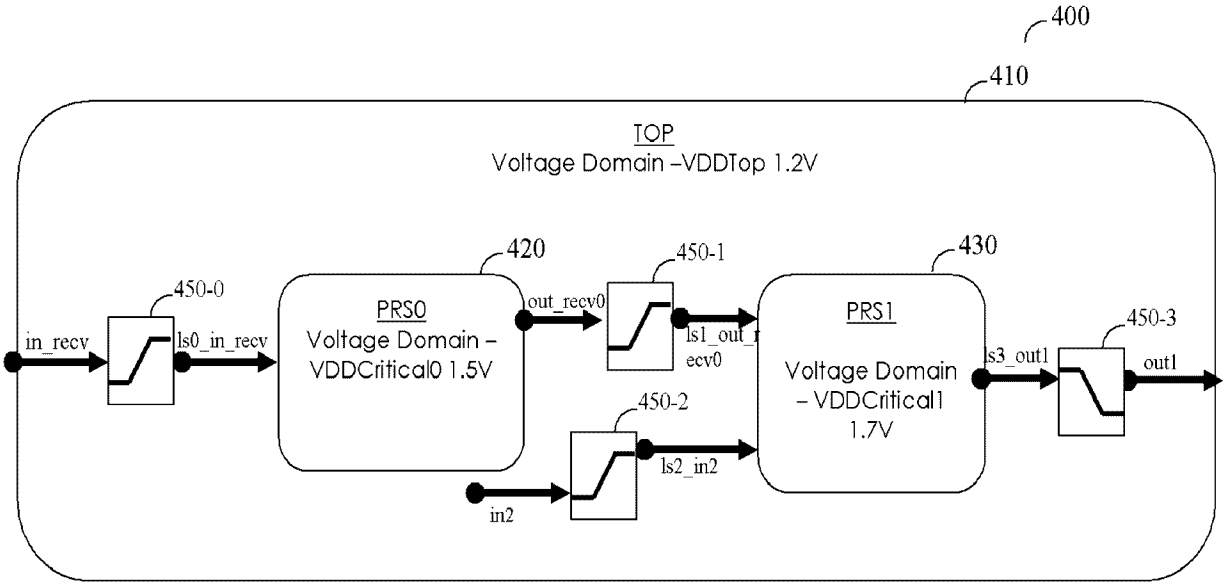
400
410
TOP
Voltage Domain –VDDTop 1.2V
420
450-1
430
450-0
PRS0
Voltage Domain –
VDDCritical0 1.5V
out_recv0
PRS1
Voltage Domain
– VDDCritical1
1.7V
450-3
in_recv
ls0_in_recv
ls1_out_r
ecv0
ls3_out1
out1
450-2
ls2_in2
in2

FIGURE 4E

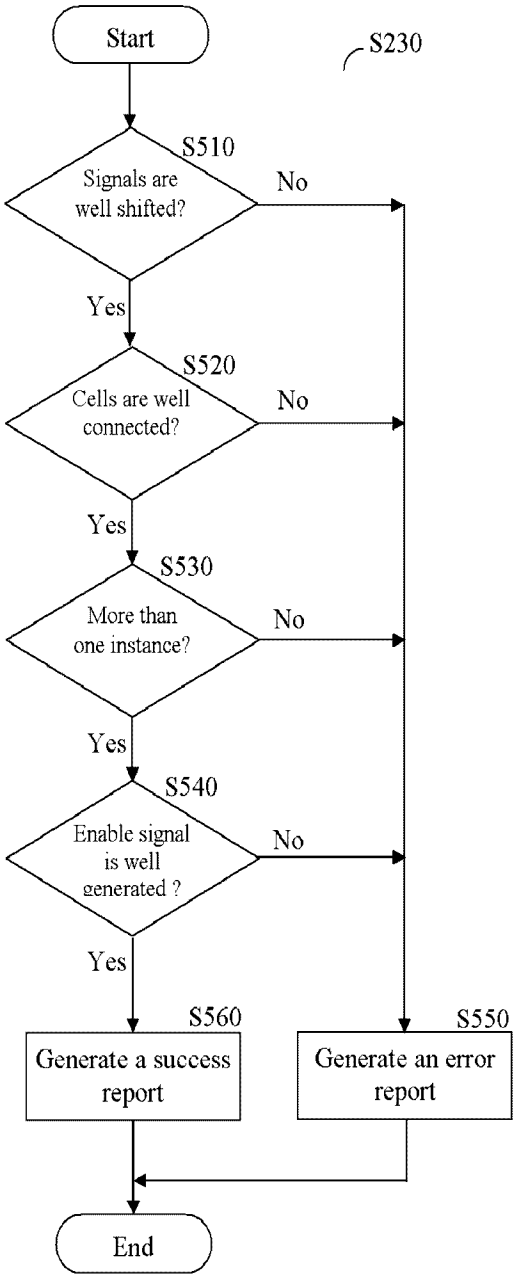
Start
S230
S510
Signals are well shifted?
No
Yes
S520
Cells are well connected?
No
Yes
S530
More than one instance?
No
Yes
S540
Enable signal is well generated ?
No
Yes
S560
Generate a success report
S550
Generate an error report
End

FIGURE 5

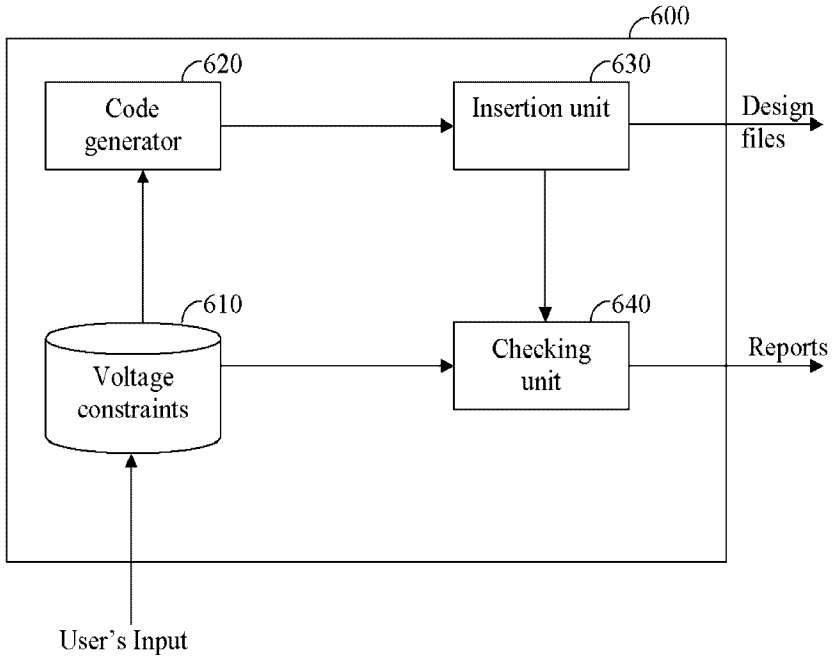
600
620
Code generator
630
Insertion unit
Design files
610
Voltage constraints
640
Checking unit
Reports
User's Input

FIGURE 6